

UML Sequence Diagram을 활용한 공대공 교전 전술 명세

박명환¹⁾ · 오지현²⁾ · 김천영²⁾ · 설현주^{*,3)}

¹⁾ 공군사관학교 전산정보학과

²⁾ 국방과학연구소 항공기술연구원

³⁾ 충남대학교 국가안보융합학부

The Specification of Air-to-Air Combat Tactics Using UML Sequence Diagram

Myunghwan Park¹⁾ · Jihyun Oh²⁾ · Cheonyoung Kim²⁾ · Hyeonju Seol^{*,3)}

¹⁾ Department of Computer Science and Information, Korea Air Force Academy, Korea

²⁾ Aerospace Technology Research Institute, Agency for Defense Development, Korea

³⁾ School of Integrated National Security, Chungnam National University, Korea

(Received 25 February 2021 / Revised 15 April 2021 / Accepted 30 September 2021)

Abstract

Air force air-to-air combat tactics are occurring at a high speed in three-dimensional space. The specification of the tactics requires dealing with a quite amount of information, which makes it a challenge to accurately describe the maneuvering procedure of the tactics. The specification of air-to-air tactics using natural languages is not suitable because of the intrinsic ambiguity of natural languages. Therefore, this paper proposes an approach of using UML Sequence Diagram to describe air-to-air combat tactics. Since the current Sequence Diagram notation is not sufficient to express all aspects of the tactics, we extend the syntax of the Sequence Diagram to accommodate the required features of air-to-air combat tactics. We evaluate the applicability of the extended Sequence Diagram to air-to-air combat tactics using a case example, that is the manned-unmanned teaming combat tactic. The result shows that Sequence Diagram specification is more advantageous than natural language specification in terms of readability, conciseness, and accuracy. However, the expressiveness of the Sequence Diagram is evaluated to be less powerful than natural language, requiring further study to address this issue.

Key Words : Manned-Unmanned Teaming Combat Tactics(유무인 협업 전술), UML Sequence Diagram(UML 시퀀스 다이어그램), Sequence Diagram Extension(시퀀스 다이어그램 확장)

1. 서론

공군의 작전은 제공작전, 전략공격작전, 전자전 등 다

* Corresponding author, E-mail: hjseol@cnu.ac.kr
Copyright © The Korea Institute of Military Science and Technology

양하게 구성되어 있지만, 모든 전술에 항공기가 참여한다는 점에서는 동일하다. 항공기가 참여하는 공군의 전술은 지상이나 해상에서의 타군 전술과는 다르게, 참가하는 전력의 속도가 매우 빠르기 때문에, 전술 구사가 분단위나 초단위로 계획되고, 구사되어야 한다. 또한, 3차원 공간에서 전술이 진행되기 때문에, 전술을 정확하게 설명(명세)하기 위해서는 참가전력의 위치, 고도, 속도, 그리고 비행 방향까지 상세히 기술 되어야 한다. 현재까지는 공군 전술을 기술하기 위해서 주로 자연어(한글 또는 영어)를 사용하고, 보조적으로 도표를 사용해서 이해를 돕는다. 하지만, 비정형적인 자연어와 도표로 명세된 공군 전술은 모호함과 부정확함이 항상 내재되어 있다. 또한, 자연어를 사용해서 전술 수행 중에 다수의 항공기간에 발생하는 지시나 메시지 등을 시간 순서로 정확히 기술하기에는 어려운 점이 많다. 근래에는 컴퓨터 기술의 발전으로, 3차원 동영상으로 시현하는 방법도 사용되지만, 3차원 동영상 작성에 많은 비용과 시간이 소모되기 때문에, 모든 전술 설명을 3차원 동영상으로 제작하는 것은 어려움이 있다. 또한, 전술에 있어서 동영상으로 표현하지 못하는 정보들이 존재한다. 가령, 동영상은 사물의 외형과 동작을 보여주기에 편리하지만, 사물의 내부적인 상태를 나타내기에는 부적합하다. 예를 들어, 공중전술에서 아군기와 적기의 위치와 기동은 3차원 동영상으로 효율적으로 묘사할 수 있지만, 전술 진행 단계나 아군기 레이더의 적기 탐지 여부 등 외부에서 시각적으로 보이지 않는 정보는 3차원 동영상으로 표현하기가 곤란하다. 또한, 3차원 동영상은 특정 개체들의 동작을 보여주는 편리하지만, 특정 시간대에 시나리오에 참여하는 모든 개체의 동작을 한꺼번에 보여주는 적절하지 않다. 따라서, 공군의 전술을 효과적으로 명세할 수 있으면서도, 표현에 모호함이 없는 정형적인 명세 방법이 필요하다.

UML의 Sequence Diagram^[1]은 소프트웨어 구성요소(object)들이 시간 진행에 따라 교환하는 메시지나 이벤트를 직관적으로 파악할 수 있게 해주는 다이어그램이다. Sequence Diagram은 객체지향 프로그램의 명세에 주로 사용되어 왔다. 하지만, Sequence Diagram이 프로그램 진행 시간에 따른 구성요소들의 행위를 기술한다는 점에서 공군전술에 참여하는 전력들의 동작을 기술하기 위해서도 효과적으로 사용할 수 있다고 판단된다. 특히, 전술 시나리오에 참여하는 전체 전력들의 상호작용을 시간 순서에 따라 쉽게 표현할 수 있다는 점이 큰 장점으로 꼽을 수 있다. Sequence

Diagram은 UML에 포함된 명세 언어로서, 1990년대 중반에 개발된 이후 ISO 표준으로 채택되었으며, 언어의 syntax와 semantics에 많은 표준이 이루어져 왔기 때문에, 언어 자체가 가지는 모호함이 없다는 장점도 있다. 또한, Sequence Diagram을 포함해서 UML의 모든 다이어그램은 syntax가 직관적이고, semantics가 명확하기 때문에 초보자라도 배우기 쉬운 장점이 있다^[2].

하지만, Sequence Diagram은 공대공 전술 명세를 위해서 개발된 것이 아니기 때문에, 기존의 Sequence Diagram만을 사용해서는 공대공 전술의 정확한 명세에 제한이 있다. 가령, 전술에 참여하는 전력의 기동(비행 속도, 방위, 고도 등)이나 상태정보(state)를 표현하는데 있어서는 한계가 있다. 따라서, 본 논문에서는 공군의 전술을 효율적으로 표현하기 위하여, 기존의 Sequence Diagram을 확장하는 방법을 제안한다. 그리고, 유인기와 무인기가 협업을 하여 적기에 대항하는 공대공 유무인 협업 전술을 이 확장된 Sequence Diagram을 사용하여 명세한 결과를 소개한다. 공대공 유무인 협업 전술은 유인기와 무인기간의 빈번한 상호작용이 필요하고, 다양한 적기를 상대해야 한다는 점에서 전술이 복잡하기 때문에, 이 전술을 Sequence Diagram이 효율적으로 설명할 수 있으면, 공군의 다른 전술들에게도 동일하게 적용할 수 있을 것으로 판단된다.

본 논문의 구성은 다음과 같다. 2장에서는 UML의 Sequence Diagram과 Sequence Diagram을 확장한 기존 연구들에 대해서 살펴본다. 그리고, 공대공 유무인 협업 전술에 대해서 소개하고, 이 전술을 기술하기 위한 Sequence Diagram의 장점과 단점에 대해서도 소개한다. 3장에서는 공대공 전술 명세에 적합하도록 Sequence Diagram을 확장하는 방법에 대해서 설명하고, 4장에서는 자연어와 확장된 Sequence Diagram을 사용하여 공대공 전술을 명세한 사례를 소개하고, 두 명세의 차이점을 비교하고, 5장에서 결론을 맺는다.

2. 배경지식

이장에서는 UML의 Sequence Diagram의 syntax와 semantics를 간단히 소개하고, Sequence Diagram의 표현력을 보강하기 위해 Sequence Diagram의 syntax와 semantics를 확장한 연구 사례를 소개한다. 또한, 유무인 협업 전술에 대해서 소개하고, 공군 전술 기술을 위한 Sequence Diagram의 장점과 단점을 소개한다.

2.1 UML Sequence Diagram 소개

UML의 Sequence Diagram은 소프트웨어의 구성요소들을 식별하고, 그들이 실행 동안에 어떤 상호작용을 하는지 기술하기 위해 사용하는 다이어그램이다. 소프트웨어의 구성요소들은 다이어그램에서 사각형으로 표시되며, 참여자(participant)라고 한다. lifeline은 참여자 사각형의 중심에서 수직 점선으로 표현되며, 참여자의 시간 흐름을 의미한다. 활성화(activation)는 lifeline 위에 직사각형 막대로 표시되며, 참여자의 활동이 직사각형 막대의 시작 지점에서 활성화되고, 마지막 지점에서 종료됨을 나타낸다. Fig. 1은 참여자, lifeline, 활성화를 보여준다.

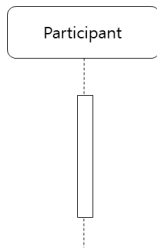


Fig. 1. Participant, lifeline and activation

메시지(Message)는 참여자 간에 데이터나 이벤트를 주고받는 것을 나타낸다. 크게 동기(synchronous) 메시지, 비동기(asynchronous) 메시지, 자체(self call) 메시지, 반환(return) 메시지로 구분할 수 있다. 동기 메시지는 요청을 보낸 후 결과가 올 때까지 대기하는 메시지를 의미하고, 비동기 메시지는 요청을 보낸 후 결과를 기다리지 않는 메시지를 의미한다. 자체 메시지는 자신에게 보내는 메시지를 의미하며, 자체적으로 업무를 처리할 때 사용한다. 반환 메시지는 동기 메시지에 대한 응답으로 결과를 전송할 때 사용한다. Fig. 2는 메시지 종류를 보여주고 있다.

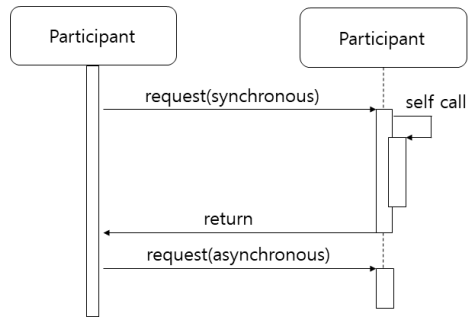


Fig. 2. Message

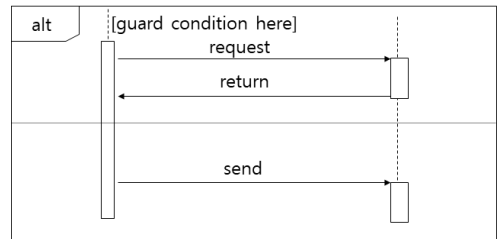


Fig. 3. Alternative flow control

Fig. 4의 option 연산자는 'opt'로 표시되고, guard condition이 만족하면 실행되는 구문을 의미한다. alternative에서 else 구문이 없다고 생각하면 된다.

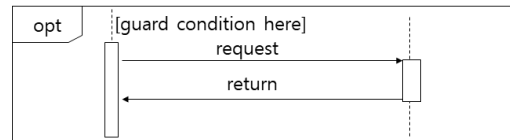


Fig. 4. Option flow control

Fig. 5의 loop 연산자는 반복을 구문에 사용하며, guard condition이 참이 되는 한 loop 안의 구문은 반복 실행된다. 일반 프로그래밍 언어의 for나 while과 동일하다.

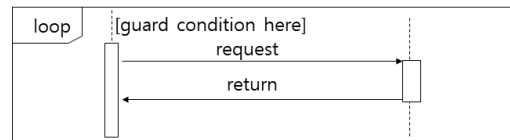


Fig. 5. Loop flow control

Sequence Diagram에서는 시간 순으로 참여자들의 상호작용을 표시하기 때문에, 상황에 따라 상호작용을 조절할 수 있는 흐름 제어(flow control)가 필요하다. Sequence Diagram에서는 대표적으로 alternative, option, loop 연산자를 통해 흐름을 제어한다. Fig. 3의 alternative(줄여서 alt)는 일반 프로그래밍 언어에서 if-else 구문을 사용하는 것과 동일하다. 조건에 사용되는 것을 guard condition이라고 하는데, 이 guard condition의 값에 따라서 두 개의 선택(if와 else) 중에 한 개가 실행된다.

2.2 UML Sequence Diagram 확장에 관한 기존 연구 Sequence Diagram의 가장 중요한 사용 목적은 시스

템의 동적인 행동을 기술하는 것이다. 하지만, 기존의 Sequence Diagram에 새로운 표기와 의미를 추가하여, 새로운 분야에 사용하려는 다양한 연구들이 진행되어 왔다.

Refsdal^[3]은 Sequence Diagram의 흐름 제어 부분에서 ‘palt’ 연산자를 추가하였다. 이 연산자는 실행해야 할 구문을 확률적으로 선택하게 하는 것이다.

Haugen^[4]은 개체들 사이에 교환되는 연속적인 이벤트(traces)들을 3가지 종류(positive, negative, inconclusive)로 분류하고, 이 이벤트들을 처리할 방법(연산자 추가)을 제안하였다.

Misbhauddin^[5]은 Sequence Diagram의 흐름 제어에 사용하는 연산자들(가령, alt, opt, loop)의 사용방법과 의미(semantics)를 구체화하기 위하여, UML 메타모델을 확장하는 방안을 제시하였다.

Silva^[6]는 다양한 개체(multi agents)들 사이의 상호작용을 명확하게 표현하기 위하여, 메시지 교환에 관한 의미를 확장하는 방안을 제시하였다.

Padilla^[7]는 동일한 종류의 개체들이 여러 개 있을 경우, 이 개체들과의 메시지교환 방식을 제시하였다.

2.3 유무인 협업 공대공 전술 사례 소개

이 절에서는 공대공 전술 명세에 관한 사례로서, 유인기와 무인기가 편대를 이루어 적을 공격하는 전술을 소개한다^[8]. 이 유무인 협업 전술에서는 유인기는 레이더를 탑재하고, 무인기는 미사일을 탑재하며, 유인기의 지휘통제 아래 무인기는 미사일을 발사한다. 이 전술의 일부분이 Table 1에 나타나 있다.

2.4 공대공 전술 명세를 위한 Sequence Diagram 장점

Sequence Diagram은 소프트웨어 명세에 주로 사용되지만, 공대공 전술 명세에도 효과적으로 사용될 수 있는 몇 가지 장점이 있다.

첫째, 공대공 전술에 참여하는 개체를 쉽고 명확하게 식별할 수 있다는 점이다. Table 1에서 소개된 자연어 설명에서는 전술에 참여하는 전력(개체)이 어떤 것들이 있는지 한눈에 보이지 않는다. 하지만, 이 전술을 Sequence Diagram으로 표현하면, 어떤 전력들이 전술에 참여하는지를 간단하게 식별할 수 있다. Fig. 6은 이 전술에 참여하는 개체들을 보여주고 있다.

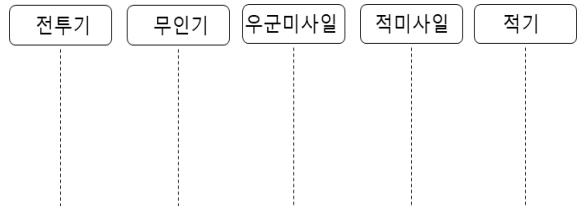


Fig. 6. Participants in the combat tactic of Table 1

둘째, 자연어로 명세된 공대공 전술은 전투에 참여하는 개체 간에 주고받는 정보가 시간 순서에 따라서 명확하게 드러나지 않는다. 공대공 전술은 분초 단위로 진행되기 때문에, 정보교환이 정확한 시간과 순서로 이루어지지 않을 경우, 전술 수행이 불가능하게 된다. 이 전술을 Sequence Diagram을 사용해서 기술하면, 개체 간의 상호작용이 시간 순서에 따라서 명확하게

Table 1. An example of the natural language specification of manned-unmanned teaming combat tactic

초기 원거리 공대공 교전은 적기 간 상호 거리가 60 NM에서 시작되어 상대방을 향해 비행하는 것으로 가정한다. 이때 무인기의 비행 형태는 우군 유인기의 전방 11 NM 전방에 위치한다. 우군 유인기와 적기간 상호거리가 55 NM에서 양쪽 레이더는 상대방을 탐지하게 된다. 그러나 적은 미사일의 최대 발사거리를 충족하지 못하여 발사는 불가능한 상황이다. 그러나 무인기는 우군 유인기 보다 11 NM 전방에 위치하고 있기 때문에 적 유인기와 무인기간의 거리는 44 NM이며 이미 적의 레이더에 탐지가 되어 있는 상황이다. 우군 유인기와 적기간의 거리 55 NM, 적기와 무인기 상호간의 거리 44 NM에 도달하면 상호 무장을 발사할 수 있는 조건이 형성된다. 무인기는 적기를 향하여, 적은 무인기를 향하여 무장을 발사한다. 이때 무인기는 탐지된 적기의 정보를 우군 유인기의 Data Link를 통하여 지속적으로 받아 무장 발사가 가능하다. 미사일의 속도는 평균속도 Mach 4.0으로 가정한다. 무인기는 전장과 반대 방향으로 최대 속도로 이탈하고 적기는 레이더 정보를 지속적으로 Update하기 위하여 미사일이 능동 유도 전환거리가 되기 전까지 전장 방향으로 비행해야 한다. 우군 유인기도 미사일 정보를 Update하기 위하여 전장 방향으로 지속 비행하여야 한다.

식별된다. Fig. 7은 Sequence Diagram에서 전술에 참여한 개체 간의 상호작용 사례를 보여주고 있다. 어떤 종류의 데이터나 이벤트가 언제 발생해야 하는지 명확하게 파악할 수 있다.

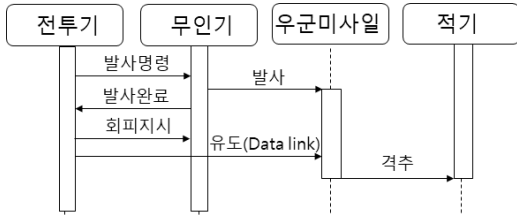


Fig. 7. Interactions of participants(sample)

셋째, 공대공 전술에 참여하는 개체들이 언제 활성화되고, 언제 종료되는지에 대한 정보가 자연어 명세에서는 불분명한데, Sequence Diagram 명세에서는 명확하게 식별된다. 가령, 아군의 미사일이 언제 활성화(발사)되고, 언제 소멸하는지, 적기는 언제 소멸하는지 등에 대한 정보들이 자연어 명세에서는 명확하지 않지만, Sequence Diagram에서는 Fig. 7에서 보는 바와 같이 한눈에 식별된다.

2.5 공대공 전술 명세를 위한 Sequence Chart 한계점

앞 절에서 설명한 바와 같이 Sequence Diagram은 공대공 전술 명세에 있어서 몇 가지 장점이 있지만, 표기법의 단순함 때문에 공대공 전술을 완벽히 표현하지 못하는 한계점도 몇가지 가지고 있다.

첫째, 전술에 참여하고 있는 개체들의 내부 상태에 대해서 표현을 할 수가 없다. Sequence Diagram에서는 개체의 활성화 여부만 표현할 수 있다. 하지만, 공대공 전술에서는 레이더의 탐지 여부나 미사일 발사 거리 등에 따라서 항공기나 미사일이 다양한 상태를 가질 수 있어야 한다.

가령, Table 2의 전술 설명에서 유인기의 레이더는 적기와와의 거리가 55 NM이 되면 탐지상태가 되어야 하고, 44 NM이 되면 미사일은 발사 가능한 상태가 되어야 한다. 하지만, 기존의 Sequence Diagram에서는 개체들의 상태를 표현하는 방법이 없다.

1) 이 전술은 국방과학연구소 발주과제인 “유무인 협업 공대공 교전을 위한 무인기 전술 및 자율기능 도출연구” 중간 보고서(2020.12월)에서 참조하였다.

Table 2. Radar state information of participants

우군 유인기와 적기 간 상호거리가 55 NM에서 양쪽 레이더는 상대방을 탐지하게 된다. 그러나 적은 미사일의 최대 발사거리를 충족하지 못하여 발사는 불가능한 상황이다. 그러나 무인기는 우군 유인기 보다 11 NM 전방에 위치하고 있기 때문에 적 유인기와 무인기간의 거리는 44 NM이며 이미 적의 레이더에 탐지가 되어 있는 상황이다. 우군 유인기와 적기간의 거리 55 NM, 적기와 무인기 상호간의 거리 44 NM에 도달하면 상호 무장을 발사할 수 있는 조건이 형성된다.

둘째, 공대공 전술에 참여하는 개체들은 비행 정보들을 가지고 있다. 가령, 항공기는 비행중에 비행방향, 비행속도, 비행고도 등의 정보들을 가지고 있다. 이런 정보들은 전술 수행에 필수적이기 때문에, 전술 설명에 반드시 포함되어야 한다. 예를 들어, Table 3의 전술 명세에서 항공기와 미사일의 비행정보(속도와 비행방향)는 Sequence Diagram에서는 표현될 수가 없다.

Table 3. Flight information of participants

미사일의 속도는 평균속도 Mach 4.0으로 가정한다. 무인기는 전장과 반대 방향으로 최대 속도로 이탈하고 적기는 레이더 정보를 지속적으로 Update하기 위하여 미사일이 능동 유도 전환 거리가 되기 전까지 전장 방향으로 비행해야 한다.

Table 4. Distance information of participants

초기 원거리 공대공 교전은 적기 간 상호 거리가 60 NM에서 시작되어 상대방을 향해 비행하는 것으로 가정한다. 이때 무인기의 비행 형태는 우군 유인기의 전방 11 NM 전방에 위치한다. 우군 유인기와 적기 간 상호거리가 55 NM에서 양쪽 레이더는 상대방을 탐지하게 된다.

셋째, 공대공 전술에서는 개체 간의 거리가 매우 중요한 요소가 된다. Table 4에서 보는 바와 같이 아군기와 적기의 거리가 레이더 탐지 여부를 결정하는 요소

가 되고, 미사일 발사 가능 여부를 결정한다. 또한, 유인기와 무인기의 거리는 전술의 효율을 결정하는 요인이 된다. 이렇게, 공대공 전술에는 개체 간의 거리를 명확하게 표현하는 것이 중요한데, Sequence Diagram에서는 이것을 표현하기가 적절하지 않다.

3. 공대공 전술 명세를 위한 Sequence Diagram 확장

이 장에서는 공대공 전술 명세를 위한 Sequence Diagram의 한계점을 극복하는 방안을 소개한다. 기존의 Sequence Diagram의 장점은 유지하면서, 공대공 전술 명세에 부족한 정보표현 방법을 추가한다.

3.1 개체들의 상태정보 표현

앞에서 설명한 것처럼 기존의 Sequence Diagram에서는, 개체의 활성화 여부에 대한 상태표현만 가능하다. 따라서 활성화 이외의 상태정보 표현 방법이 필요한데, 이것을 위하여 Sequence Diagram의 활성화(activation) 표기에 상태정보를 추가하는 방안을 제시한다. Fig. 8은 유인기 개체의 활성화 표기에 레이더 상태정보를 추가한 것을 보여주고 있다. 유인기는 적기와와의 거리가 55 NM 밖에 있으면, 미탐지 상태에 있고, 적기가 55 NM 이내에 들어올 때 탐지상태로 변한다. 그리고 44 NM 이내에 들어오고, 미사일을 발사하면 유도상태로 변한다.

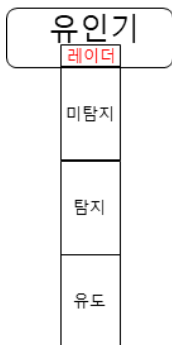


Fig. 8. Notation of state information

3.2 개체들의 비행정보 표현

공대공 전술에 참여하는 개체들의 비행 정보도 앞에서 설명한 상태정보 표현방식으로 해결할 수 있다.

비행 정보는 주로 전술에 참여하는 개체들의 비행 방향, 고도, 속도가 중요 구성 요소이다. 이 정보들을 그룹으로 묶어서 하나의 상태로 만들 수 있다. 그리고 이 구성 요소 중에서 한 개라도 내용이 바뀌면 새로운 상태로 전이하게 된다. Fig. 9에서 무인기와 유인기의 비행상태 정보를 보여주고 있다. 무인기는 레이더를 장착하지 않기 때문에 비행정보를 표현하는 비행상태만 있다. 이 전술에서 고도는 중요하지 않기 때문에, 방향과 속도만으로 비행상태를 구성하게 된다. 하나의 상태(직진, M0.9)에서 비행방향이 바뀌게 되면 새로운 상태(회피기동, M0.9)가 된다. 그리고, 최종 상태(재진입, M0.9)로 변경된다. Fig. 9에서 유인기는 레이더를 장착하고 있어서, 레이더 상태와 비행상태를 동시에 가지고 있는데, 두 개의 상태를 같이 표현하는 방법을 보여주고 있다.

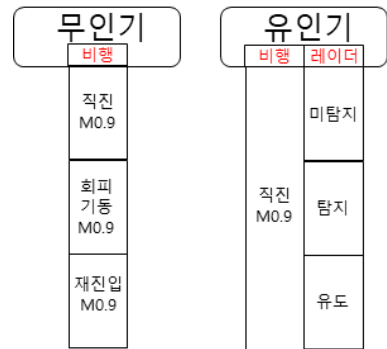


Fig. 9. Notation of flight information

3.3 개체들의 거리정보 표현

공대공 전술에 참여하는 개체 간의 거리는 시간에 따라서 바뀌게 된다. 그리고 거리에 기반해서 개체들의 상태정보와 비행정보가 변화한다. 개체 간의 거리는 해당 개체들의 lifeline에 수평 화살표를 그리고 그 화살표 위에 거리를 표시하는 것으로 나타낸다. Fig. 10은 개체 간의 거리를 붉은색 화살표와 푸른색 화살표로 나타내고 있는 것을 보여주고 있다. 푸른색 화살표는 개체 간에 일정시간 지속되는 거리를 의미한다. 즉, 유인기와 무인기는 10 NM 거리를 일정하게 유지하면서 비행하고 있다는 의미가 된다. 거리표시 밑에 거리유지의 종료시점을 대괄호안에 명시한다. 붉은 색의 화살표는 순간적인 거리를 의미한다. 무인기와 적기는 서로 접근하다가 어느 순간에 55 NM의 거리가 되면, 무인기와 적기의 거리가 55 NM 이라는 이벤트

가 발생한다. 적기의 레이더는 이 이벤트를 받으면, 미탐지에서 탐지로 상태정보가 변경된다. 유인기와 적기의 거리가 55 NM이 되면 동일하게 이벤트가 발생되고, 유인기의 레이더는 미탐지에서 탐지상태로 변경된다. 무인기와 적기의 거리가 44 NM이 되면, 무인기와 적기는 서로 간에 미사일을 발사할 수 있는 거리가 된다. 이렇게, 개체 간의 거리를 화살표로 표현하고, 이벤트 발생으로 해석한다.

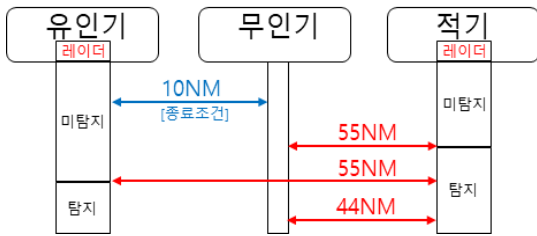


Fig. 10. Notation of the distance between participants

3.4 개체의 lifeline 종료 이유 명시

공대공 전술에 참여하는 개체들은 상황에 따라서, lifeline이 종료될 수 있다. 예를 들어서, 항공기가 격추되거나, 미사일이 표적을 상실하여 더 이상 유도가 불가능할 경우에는 전술 시나리오에서 존재할 필요가 없어진다. 이런 경우에는 lifeline 종료 마지막 부분에 대괄호를 사용하여 종료 이유를 명시한다.

4. 유무인 협업 공대공 전술의 Sequence Diagram 명세

이 장에서는 앞장에서 설명한 확장된 Sequence Diagram을 사용하여 유무인 협업 공대공 전술 사례를 명세한 결과를 소개한다. 먼저, 유무인 협업 전술의 자연어 명세를 소개하고, 동일한 전술을 확장된 Sequence Diagram로 명세한다. 그리고 자연어 명세와 Sequence Diagram 명세의 간결성과 명확성 등의 관점에서 비교 분석한다.

4.1 유무인 협업 공대공 전술 자연어 명세²⁾

유무인 협업 공대공 전술은 미래전장에서 유인기와

2) 이 전술은 국방과학연구소 발주과제인 “유무인 협업 공대공 교전을 위한 무인기 전술 및 자율기능 도출연구” 중간 보고서(2020.12월)에서 참조하였다.

무인기가 협업을 하면서 임무를 수행하는 전술을 의미한다. 이 전술에서는 유인기는 무인기의 지휘통제 임무와 무인기가 발사한 미사일을 적기까지 유도하는 임무를 수행한다. 무인기는 유인기 전방에 위치해서 미사일 발사 임무를 수행한다. 이 전술 시나리오에 참여하는 전력으로, 적기는 MIG-29급 2대이고, 아군은 유인기 1대(F-16급)와 공대공 미사일 발사기능을 갖춘 소형 무인기 2대이다. 적·아 유인기는 동일한 성능의 레이더를 장착하여 공중 탐지 능력 및 미사일 발사를 위한 정보제공 능력을 가진다고 가정하며, 장착 레이더의 탐지범위는 55 NM이다. 공대공 레이더 유도 미사일의 최대 유효 발사거리는 44 NM로 적·아 동일한 것으로 가정한다. 또한, 미사일이 항공기 레이더의 유도없이 표적을 자체적으로 추적하는 시기, 즉 active homing 전환거리는 표적과 미사일간의 거리 11 NM 전방으로 적용한다. 전술에 대한 상세한 설명은 Table 5에 나타나 있다.

4.2 유무인 협업 공대공 전술의 Sequence Diagram 명세

이 절에서는 Table 5에서 자연어로 명세한 유무인 협업 공대공 전술에 대해서, Sequence Diagram을 사용하여 명세한 결과를 소개한다. Fig. 11은 전술에 대한 Sequence Diagram 명세 결과를 보여주고 있다. 전술에 참여하는 각 개체의 Sequence Diagram 설명은 다음과 같다.

4.2.1 유인기

유인기는 ‘비행’ 상태와 ‘레이더’ 상태를 가지고 있다. ‘비행’ 상태는 방향과 속도로 구성되는데, 적기방향으로 비행을 하고, 속도는 M0.9이다. ‘레이더’ 상태는 적기와와의 거리를 기준으로 55 NM 밖에서는 ‘미탐지’ 상태가 되고, 55 NM이 되면 ‘탐지’ 상태가 된다. 44 NM이 되면 무인기에게 미사일 ‘발사명령’을 내리고, 미사일 발사가 완료되면 레이더는 ‘미사일 유도’ 상태가 된다. 적기의 미사일 발사 사실을 인지한 이후에 무인기에게 ‘회피지시’를 내린다. 그리고 적기도 회피 기동을 하게 되면 레이더가 ‘미탐지’ 상태로 바뀌게 되고, 유도하던 미사일은 유도상실로 사라진다. 적기가 회피기동을 중지하고, 전장에 재진입하면, 레이더는 다시 ‘탐지’ 상태가 되고, 무인기에 ‘재진입지시’를 내린다. 그리고 미사일 ‘발사명령’을 내리고, 발사가 완료되면 ‘미사일 유도’ 상태가 된다. 미사일이

‘자체유도’ 상태가 되어서, 더 이상의 유도가 필요없으면 ‘탐지’ 상태로 전환된다.

4.2.2 무인기

무인기는 ‘비행’ 상태만 가지고 있다. 무인기는 유인기와 10 NM 거리를 두고 비행한다. 이 거리는 무인기가 회피기동을 수행할 때까지 일정하게 유지된다. 무인기는 초기에 적기방향으로 M0.9 속도로 비행을 하다가, 유인기의 미사일 ‘발사명령’에 미사일을 발사하

고, 유인기의 ‘회피지시’에 따라서 회피기동을 수행한다. ‘재진입지시’에 따라서 전장에 재진입하고, 유인기 지시에 따라 미사일을 재발사하고, 전장을 이탈한다.

4.2.3 아군 및 적군 미사일

미사일은 ‘비행’과 ‘유도’ 상태를 가지고 있다. ‘비행’ 상태는 표적방향으로 M4.0의 속도로 비행을 한다. ‘유도’ 상태는 미사일이 누구에 의해서 유도되는지를 나타낸다. ‘미유도’는 발사 초기 가속단계에서 유도없

Table 5. The natural language specification of manned-unmanned teaming combat tactic

초기 원거리 공대공 교전은 적기 간 상호 거리가 60 NM에서 시작되어 상대방을 향해 비행하는 것으로 가정한다. 이때 무인기의 비행 형태는 우군 유인기의 전방 11 NM 전방에 위치한다. 우군 유인기와 적기간 상호거리가 55 NM에서 양쪽 레이더는 상대방을 탐지하게 된다. 그러나 적은 미사일의 최대 발사거리를 충족하지 못하여 발사는 불가능한 상황이다. 그러나 무인기는 우군 유인기 보다 11 NM 전방에 위치하고 있기 때문에 적 유인기와 무인기간의 거리는 44 NM이며 이미 적의 레이더에 탐지가 되어 있는 상황이다. 우군 유인기와 적기간의 거리 55 NM, 적기와 무인기 상호간의 거리 44 NM에 도달하면 상호 무장을 발사할 수 있는 조건이 형성된다. 무인기는 적기를 향하여, 적은 무인기를 향하여 무장을 발사한다. 이때 무인기는 탐지된 적기의 정보를 우군 유인기의 Data Link를 통하여 지속적으로 받아 무장 발사가 가능하다. 미사일의 속도는 평균속도 Mach 4.0으로 가정한다. 무인기는 전장과 반대 방향으로 최대 속도로 이탈하고 적기는 레이더 정보를 지속적으로 Update하기 위하여 미사일이 능동 유도 전환거리가 되기 전까지 전장 방향으로 비행해야 한다. 우군 유인기도 미사일 정보를 Update하기 위하여 전장 방향으로 지속 비행하여야 한다. 적기인 MIG-29가 우군의 레이더 탐지를 레이더 경보 수신기(RWR, Radar Warning Receiver)를 통하여 인지하고 회피기동을 수행한다. 적은 전장을 이탈하지 않는 이상 자신의 레이더 탐지 범위(통상 좌, 우 60도)를 유지하면서 우군 레이더의 탐지를 거부하는 기동을 수행한다. 우군의 발사된 미사일은 적의 회피기동으로 인하여 무력화되며 적 유인기는 우군의 레이더 탐지가 없다고 판단될 경우(약 10초 직진비행으로 가정) 무장 발사를 위한 전장 재진입을 시도한다. 무인기도 RWR을 장착하고 있으며 적의 레이더 탐지 및 미사일 발사 신호를 수신하게 되면 유인기와 연결된 Data Link를 통하여 위협 정보를 송신하고 유인기 조종사는 회피를 명령하게 된다. 적 유인기와 동일하게 레이더 추적 및 미사일의 유도가 불가능하도록 기동을 수행한 이후 재발사를 위하여 전장 재진입을 위하여 기동한다. 무인기는 무장 발사 이후 전장을 이탈하도록 명령이 주어졌지만, 적의 회피로 레이더 탐지가 거부되어 발사된 미사일을 유도할 수 없는 상황이 되면 유인기 조종사는 무인기의 전장 재진입을 명령할 수 있는 것이다. 이때 무인기는 재전장 진입을 위하여 무인기 최대 선회율로 선회하여 적이 있는 방향으로 기수를 향한다. 적 유인기가 무장 발사를 위하여 기수를 무인기를 향하기 이전에 지속적으로 적의 유인기를 추적중인 우군 유인기의 레이더에 의하여 무인기가 먼저 무장 발사 가능성이 주어진다. 무인기는 미사일 발사 이후 전장에서 멀어지는 방향으로 선회하여 전장이탈을 시도한다. 적기도 조건이 되면 무인기를 향해 미사일을 발사한다. 무인기는 전장과 반대 방향으로 최대 속도로 이탈하고 있는데 반하여, 적기는 레이더 정보를 지속적으로 Update하기 위하여 미사일이 능동 유도 전환거리가 되기 전까지 전장 방향으로 비행해야 한다. 우군 유인기도 미사일 정보를 Update하기 위하여 전장 방향으로 지속 비행하여야 한다. 이 단계에서 적기의 레이더는 무인기와 자신의 상대 속도에 의거 레이더의 추적이 어렵게 될 수 있다. 만일 지속 추적이 가능하다고 해도 계산에 의거 적의 미사일이 무인기에 도착하기 전 11 NM 정도에서 적기는 무인기의 미사일에 격추된다. 적기의 격추로 인하여 아직 능동 유도 단계에 진입하지 못한 적 미사일은 유도 불가능한 상태가 되어 무력화될 것이다.

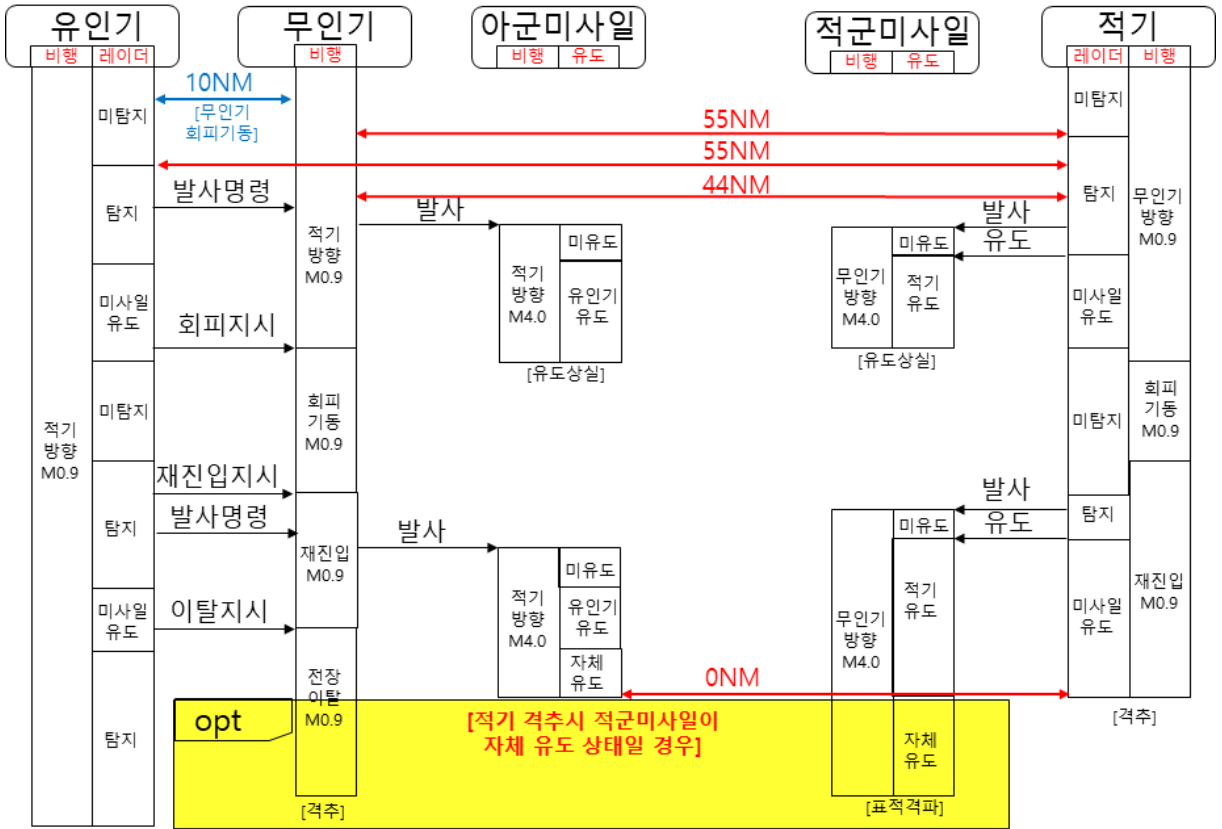


Fig. 11. The Sequence Diagram specification of manned-unmanned teaming combat tactic

이 비행하는 단계이고, ‘유인기 유도’나 ‘적기 유도’는 유인기나 적기에게서 유도를 받는 단계이다. 그리고 ‘자체 유도’는 표적과의 거리가 11 NM 이내가 되면 미사일 자체의 센서로 표적까지 유도를 하는 상태를 의미한다. 유인기나 적기로부터 유도를 받는 상태에서 유도신호가 끊어지면 미사일은 표적을 상실하고 추락해서, 전술 시나리오에서 사라진다.

4.2.4 적기

적기도 유인기와 동일하게 ‘비행’과 ‘레이더’ 상태를 가지고 있다. 레이더 상태 전환은 유인기와 동일하다. 다만, 적기는 무인기를 가지고 있지 않기 때문에, 직접 미사일을 발사하고, 유도하게 된다.

4.2.5 ‘opt’ 흐름제어 부분

Fig. 11의 노란색 부분은 option 흐름제어 부분이다. 이 부분은 조건에 따라서 실행 여부가 결정되는 부분

이다. 실행조건은 적기 격추시 미사일의 ‘유도상태’에 의해서 결정된다. 적기가 격추 되었을 때, 적 미사일의 상태가 ‘자체유도’일 경우에는, 적기의 유도 없이도 미사일 자체 센서로 비행해서 무인기를 격추하게 된다. 반대로, 적기 격추시 적 미사일의 상태가 ‘적기 유도’라면, 미사일은 유도를 받지 못하기 때문에 표적을 상실하고, 무인기는 무사히 탈출할 수 있게 된다.

4.3 자연어와 Sequence Diagram 명세의 비교

Table 5와 Fig. 11은 동일한 공대공 전술에 대해서 다른 표기법을 사용해서 명세한 것이다. 이 절에서는 이 두 표기법의 차이점에 대해서 비교 분석한다.

4.3.1 가독성 측면

가독성은 어떤 표현에 대해 읽기 측면에서 이해하기 쉬운 정도를 의미한다. 가독성이 높으면 이해하기 쉽고, 가독성이 낮으면 이해가 어렵다는 의미가 된다.

UML 언어의 가장 큰 장점 중의 하나는 이 언어의 다이어그램으로 작성된 명세의 가독성이 좋다는 것이다^[8]. Sequence Diagram에서는 공대공 전술에 참여하는 전력들이 한눈에 파악이 되며, 상호간에 교환하는 데이터와 시간진행에 따른 전력들의 행위를 일목요연하게 확인할 수 있다. 자연어로 명세한 전술에서는 Sequence Diagram에서 쉽게 확인할 수 있는 이런 정보들을 파악하기가 간단하지 않다.

4.3.2 간결성 측면

간결성은 어떤 대상을 표현함에 있어서 함축성 정도를 의미한다. 표현 도구의 간결성이 좋다는 것은 짧고 효율적으로 표현할 수 있다는 의미가 된다. Sequence Diagram을 포함하고 있는 UML은 단순하고 직관적인 표기법을 사용하고 있어서 언어자체가 매우 간결하다^[9]. 공대공 전술 표현에 있어서, 자연어와 Sequence Diagram 명세에서, Sequence Diagram 명세가 더 간결하다는 것은 Table 5와 Fig. 11을 비교하면 확인할 수 있다. 상대적으로 적은 분량으로 전술을 표현하고 있다. Sequence Diagram은 공대공 전술을 기술할 때, 정해진 프레임워크(가령, 개체, lifeline, 활성화, 메시지 교환 등)가 존재하기 때문에, 이 프레임워크에 따라서 명세하게 되면 군더더기가 없는 간결한 표현이 가능하다.

4.3.3 명확성 측면

명확성은 모호성과 반대되는 개념으로, 어떤 표현의 의미가 모든 사람들에게 동일하게 이해가 되면 명확성이 있는 것이고, 사람마다 다르게 이해가 되거나, 이해 자체가 되지 않으면 명확성이 없는 것이다. 자연어는 근본적으로 모호성이 많은 표현 수단이다. 소프트웨어 명세를 위해서 모호성이 없는 표현 도구가 필요한데, 이것이 UML의 장점이다^[10]. 따라서 수학적 의미론을 기반으로 한 UML의 Sequence Diagram이 자연어보다는 표현측면에서 명확하다고 할 수 있다.

4.3.4 표현력 측면

표현력은 복잡하거나 어려운 개념을 쉽게 나타낼 수 있는 정도를 의미한다. 표현력은 앞에서 설명한 간결성과 대조되는 측면이 있는데, 언어가 간결하면 표현력이 떨어지는 경우가 많다. Sequence Diagram은 자연어보다 매우 제한적인 표현 도구를 가지고 있어서, 자연어로는 명세할 수 있지만 Sequence Diagram에서

는 표시할 수 없는 내용이 존재한다. 예를 들면, 아래는 자연어로 명세된 전술의 일부본이다.

적의 회피로 레이더 탐지가 거부되어 발사된 미사일을 유도할 수 없는 상황이 되면 유인기 조종사는 무인기의 전장 재진입을 명령할 수 있는 것이다.

이 부분을 Sequence Diagram으로 표현하면, 유인기 개체에서 무인기 개체로 ‘재진입 지시’ 메시지만 나타낼 수 있고, 상황설명에 대한 표현은 불가하다.

4.3.5 확장성/유연성 측면

확장성/유연성은 명세의 내용에 변경이 필요할 경우, 얼마나 쉽게 그 변경이 가능한지에 대한 변경의 용이성을 의미한다. Sequence Diagram은 자연어 표기법에 비해 확장성/유연성이 떨어진다. 기존의 전술에 변경이 필요할 경우, 자연어 표기에서는 변경내용을 그대로 기술하면 되지만, Sequence Diagram에서는 변경내용이 영향을 미치는 모든 곳의 다이어그램을 수정해주어야 한다. 가령, 앞에서 설명한 공대공 전술에서 아군 레이더의 탐지거리가 기존의 55 NM에서 70 NM로 변하게 되면, Table 5의 자연어로 표기된 공대공 전술에서는 55 NM을 70 NM로 변경하고, 이에 따른 시나리오를 수정하면 되기 때문에 상대적으로 변경이 쉽다. 하지만, 이 변경을 Fig. 11의 Sequence Diagram에 적용하게 되면, 다이어그램 전체의 변경이 필요하게 되어서, Sequence Diagram은 명세 변경에 대한 확장성과 유연성이 떨어진다고 할 수 있다.

Table 6은 공대공 전술 기술에 있어서, 자연어 표기법과 Sequence Diagram의 장점과 한계점을 정리한 것을 보여주고 있다. Table 6에서 보는 것처럼 본 논문에서 제시한 Sequence Diagram의 확장은 자연어 표기의 장점을 완전히 대체하지 못하고 있다. 또한, 공대공 전술을 명세하기 위해 자연어와 함께 사용하는 그림 도표도 Sequence Diagram이 완전히 대체하지 못한다. Fig. 12는 공대공 전술을 기술하고 있는 그림 도표의 사례를 보여주고 있다. 이 전술은 유인기와 무인기가 협업을 하여, 적기를 상대하는 전술을 보여주고 있는데, 항공기들의 비행궤적은 그림 도표를 사용하여야만 명확하게 표현할 수 있다. Sequence Diagram을

사용해서는 3차원 공간에서의 항공기 비행경로를 효율적으로 표현할 수가 없다. 따라서 공대공 전술의 완전하고, 정확한 명세를 위해서는 본 논문에서 소개한 확장된 Sequence Diagram과 함께, 자연어와 그림 도표를 상호보완적으로 같이 사용할 필요가 있을 것으로 판단된다.

Table 6. The comparison between natural language notation and Sequence Diagram notation

평가측면	자연어 표기	Sequence Diagram
가독성	낮음	높음
간결성	낮음	높음
명확성	낮음	높음
표현력	높음	낮음
확장성/유연성	높음	낮음

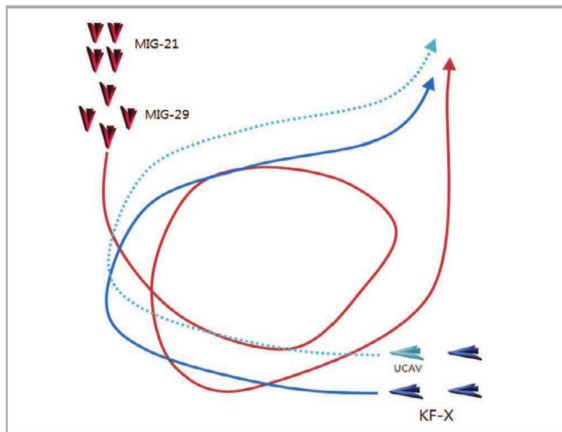


Fig. 12. A figure depicting air-to-air combat tactics^[11]

5. 결론

공대공 전술은 3차원 공간에서 빠른 속도로 발생하기 때문에, 매우 다양한 정보들의 정확한 기술이 필요하다. 자연어를 사용한 공대공 전술의 명세는 모호성과 가독성 부족으로 불리한 점이 많다. 따라서 본 논문에서는 공대공 전술 명세를 위해 수학적 의미론을 가진 UML의 Sequence Diagram을 사용하는 방법을 제

시하였다. 기존의 Sequence Diagram으로는 공대공 전술을 모두 표현할 수가 없으므로, Sequence Diagram 표기법의 확장방안을 제안하였다. 사례연구로서, 확장된 Sequence Diagram을 사용하여, 유무인 협업 공대공 전술을 명세한 내용을 소개하였다. 그리고 자연어 명세와 Sequence Diagram 명세 내용을 가독성, 간결성, 정확성, 표현성, 확장성/유연성 측면에서 비교 분석하였다. 가독성, 간결성, 정확성 측면에서 Sequence Diagram 표현이 자연어 표현보다 유리하였다. 하지만, 자연어로는 쉽게 표현할 수 있는 공대공 전술의 배경 내용이나 상황설명을 Sequence Diagram에서는 불가능하였다. 또한, 전술 변경에 대한 확장성과 유연성 측면에서 Sequence Diagram은 부족함이 드러났다. 따라서 Sequence Diagram의 표현력과 확장성/유연성을 추가로 개선하는 방안에 관한 후속 연구가 필요하다. 또한, 본 논문에서 제시한 Sequence Diagram의 확장 기능을 UML 자동화 도구에 추가할 필요가 있다. 본 논문에서 제시한 Sequence Diagram 확장기능은 참여자(participant)의 상태정보와 거리정보를 추가하는 것이기 때문에 구현이 복잡할 것으로 보이지 않는다. 기존에 개발되어 있는 Sequence Diagram 오픈소스 도구가 이 기능을 추가하는 것도 향후 과제가 될 것이다.

후 기

본 연구는 국방과학연구소의 지원(UD200021JD)에 의해 수행되었으며, 이에 감사드립니다.

References

- [1] Fowler. M, "UML Distilled, Third Edition," The Addison-Wesley, 2003.
- [2] SCHMULLER. J, "Teach Yourself UML in 24 Hours," Indianapolis, SAMS, 1999.
- [3] Refsdal. A and Stølen. K, "Extending UML Sequence Diagrams to Model Trust-dependent Behavior With the Aim to Support Risk Analysis," Electronic Notes in Theoretical Computer Science, Vol. 197, No. 2, pp. 15-29, 2008.
- [4] Haugen. Ø, Husa.K. E, Runde. R. K, and Stølen. K, "STAIRS Towards Formal Design with Sequence

- Diagrams,” Software and Systems Modeling, Vol. 4, No. 4, pp. 355-357, 2005.
- [5] Misbhauddin. M and Alshayeb. M, “Extending the UML Metamodel for Sequential Diagram to Enhance Model Traceability,” Proceedings of the Fifth International Conference on Software Engineering Advances, 2010.
- [6] Silva. V. T and Lucena. C. J, “Extending the UML Sequence Diagram to Model the Dynamic Aspects of Multi-Agent Systems,” Rio de Janeiro, Brazil, MCC 15/03, 2003.
- [7] Padilla. G, Serrano. M. A, and Oca. C. M, “A UML Sequence Diagram Extension to Handle Multiplicities,” Proceedings of the Fifth Mexican International Conference in Computer Science, 2004.
- [8] Ketema K. G, “Challenges and Opportunity of UML Diagram for Software Project Development as a Complete Modeling Tool,” IOSR-JMCA, Volume 7, pp. 46-48, 2020.
- [9] Miles R and Hamilton K, “Learning UML 2.0,” O’Reilly Online Learning Platform, 2006.
- [10] Unnati. S. S and Devesh C. J, “Resolving Ambiguity in Natural Language Specification to Generate UML Diagrams for Requirements Specification,” International Journal of Software Engineering, Technology and Application, Vol. 1, pp. 308-334, 2015.
- [11] Kim J, Byun Y, Jo B, “The Study of Operation Concept and Development Direction of Air-to-Air Unmanned Fighter Using COBRA Model,” KIDA Defense Analysis(Korean Edition), Vol. 1625, 2016.